

Amendments to the Claims:

1. (Currently amended) A computer system for generating metadata for use during stack unwinding, comprising:

a processor; and

a memory having computer-executable instructions stored thereon, wherein the computer executable instructions include:

a plurality of procedures, wherein each procedure comprises a sequence of binary instructions;

a runtime for generating unwind data, wherein the unwind data includes a first plurality of blocks of metadata having a first order of blocks, wherein each block of metadata is associated with a corresponding procedure in the plurality of procedures, wherein each block of metadata in the plurality of blocks of metadata includes at least one unwind table and at least one unwind information block; and

an unwind rewriter programmed to obtain the unwind data and reorder the first plurality of blocks of metadata to generate a second plurality of blocks of metadata having a second order, wherein the unwind rewriter reorders the first plurality of blocks in accordance with a second unwind table and a second unwind information block, wherein reordering the first plurality of blocks in accordance with a second unwind table and a second unwind information block further comprises determining when basic blocks identified in a single unwind table associated with the first order of blocks are associated with more than one unwind table associated with a current order of basic blocks, ~~and~~ creating a new region header describing a region of zero length when the basic blocks identified in the single unwind table associated with the first order blocks are associated with more than one unwind table associated with the current order of basic blocks, and copying non-when action descriptor records from the unwind information block into the second unwind information block when the basic blocks identified in the single unwind table associated with the first order blocks are associated with more than one unwind table associated with the current order of basic blocks, wherein the first plurality of blocks are reordered in response to a modification of the sequence of binary instructions within a

procedure, such that the second plurality of blocks of metadata accurately represents the same runtime semantics as that of the unmodified sequence of binary instructions.

2. (Cancelled)

3. (Previously presented) The computer system of claim 1, wherein the at least one unwind information block includes a region header describing a region of zero length.

4. (Currently amended) A computer-implemented method of regenerating unwind data for a modified binary procedure wherein a current order of basic blocks within the modified binary procedure differs from an original order of the basic blocks, the computer-implemented method comprising:

obtaining original unwind data that describes the original order of the basic blocks, wherein the original unwind data is associated with an unwind table and unwind descriptor records, wherein the original order of basic blocks is noncontiguous and the original unwind data describes the noncontiguous order;

regenerating new unwind data from the original unwind data, wherein regenerating new unwind data includes generating new unwind tables and new unwind descriptor records, wherein each new unwind table pertains to a different grouping of contiguous basic blocks within the noncontiguous original order of basic blocks, wherein the new unwind data includes a reordering of the original order of basic blocks, wherein regenerating the new unwind descriptor records further comprises determining when basic blocks identified in a single unwind table associated with the original order of basic blocks are associated with more than one unwind table associated with the current order of basic blocks, and creating a new region header describing a region of zero length when the basic blocks identified in the single unwind table associated with the original order of basic blocks are associated with more than one unwind table associated with the current order of basic blocks, and wherein the reordering represents the same runtime semantics as that of the unmodified sequence of binary instructions, and copying non-

when action descriptor records from the original unwind descriptor records into the new unwind descriptor records; and

writing the new unwind data to the modified binary procedure.

5. (Previously presented) The method of claim 4, wherein obtaining the unwind data comprises parsing the original unwind data that describes the original order of the basic blocks.

6. (Original) The method of claim 5, wherein parsing the original unwind data comprises identifying a start basic block and an end basic block of a region associated with the modified binary procedure.

7. (Original) The method of claim 6, wherein identifying the end basic block of the region further comprises splitting a single basic block into two basic blocks, such that a first basic block ends on a last instruction of the region.

8. (Original) The method of claim 6, wherein parsing the original unwind data further comprises identifying an unwind information block associated with a basic block in the original order of the basic blocks that includes a when action description record and establishing a link between the when action description record and the corresponding instruction in the basic block.

9.-10. (Cancelled)

11. (Currently amended) A computer-implemented method for regenerating unwind data in response to a binary modification to a procedure, the procedure including a plurality of basic blocks, comprising:

receiving unwind data comprising an unwind table and a plurality of unwind descriptor records wherein the unwind data is associated with a procedure having binary instructions;

modifying the procedure to perturb the binary instructions of the procedure;
parsing the unwind data to identify a start basic block and an end basic block for a region associated with the procedure; and

rewriting the unwind data, wherein the rewriting of unwind data includes a reordering of unwind data in accordance with a second unwind table and a second plurality of unwind descriptor records such that the rewritten unwind data accurately represents the runtime semantics of the binary instructions before the binary instructions were perturbed, wherein generating the second plurality of unwind descriptor records further comprises determining when basic blocks identified in a single unwind table associated with the unmodified procedure are associated with more than one unwind table associated with the binary modified procedure; and, ~~and creating a new region header describing a region of zero length when the basic blocks identified in the single unwind table associated with the unmodified procedure are associated with more than one unwind table associated with the binary modified procedure.~~

when basic blocks identified in the unwind table are associated with more than one unwind table, creating a new region header describing a region of zero length and copying non-when action descriptor records from the unwind descriptor records into the second unwind descriptor records.

12. (Previously presented) The computer-implemented method of claim 11, wherein parsing the unwind data further comprises generating a relationship between a when-descriptor within an unwind descriptor record and an instruction in the procedure, and rewriting the unwind data further comprises associating the when-descriptor with an appropriate unwind descriptor record in the second plurality of unwind descriptor records.

13. (Original) The method of claim 11, wherein identifying the end basic block of the region further comprises splitting a single basic block into two basic blocks, such that a first basic block ends on a last instruction of the region.

14.-18. (Cancelled)

19. (Currently amended) A computer-readable storage medium having computer-executable instructions for rewriting unwind data in response to a binary modification to a procedure, the procedure including a plurality of basic blocks, the instructions comprising:

- receiving unwind data comprising an unwind table and a plurality of unwind descriptor records wherein the unwind data is associated with a procedure having binary instructions;
- modifying the procedure to perturb the binary instructions of the procedure;
- parsing the unwind data to identify a start basic block and an end basic block for a region associated with the procedure; ~~and~~
- rewriting the unwind data, wherein the rewritten unwind data includes a reordering of the unwind data according to a second unwind table and a second plurality of unwind descriptor records such that the rewritten unwind data accurately represents the runtime semantics of the binary instructions before the binary instructions were perturbed, wherein reordering of the unwind data according to a second unwind table and a second plurality of unwind descriptor further comprises determining when basic blocks identified in the unwind table are associated with more than one unwind table associated with the binary modified procedure; ~~and, and creating a new region header describing a region of zero length when the basic blocks identified in the unwind table associated with the unmodified procedure are associated with more than one unwind table associated with the binary modified procedure.~~

when basic blocks identified in the unwind table are associated with more than one unwind table, creating a new region header describing a region of zero length and copying non-when action descriptor records from the unwind descriptor records into the second unwind descriptor records.

20. (Currently amended) A computer-readable storage medium having computer-executable instructions for rewriting unwind data wherein a current order of basic blocks within the modified binary procedure differs from an original order of the basic blocks, the instructions comprising:

obtaining original unwind data that describes the original order of the basic blocks, wherein the original unwind data is associated with an unwind table and unwind descriptor records, wherein the original order of basic blocks is noncontiguous and the original unwind data describes the noncontiguous order;

rewriting the original unwind data, wherein the rewritten unwind data includes a reordering of the original order of basic blocks, wherein rewriting the original unwind data includes:

parsing the original unwind data to identify a start block and an end block for region headers associated with the procedures in the modified binary procedures, wherein the identified start block and the identified end block are recorded in a procedure side table,

recording when action description records in the procedure side table,

generating new unwind tables ~~a new unwind table~~ based in the procedure side table, wherein each new unwind table pertains to a different grouping of contiguous basic blocks within the noncontiguous original order of basic blocks,

generating new unwind descriptors based on the procedure side table and the new unwind table,

reordering the original unwind data according to the new unwind table and the new unwind descriptors, wherein reordering includes determining when basic blocks identified in the new unwind table are associated with more than one unwind table, ~~and creating a new region header describing a region of zero length when the basic blocks identified in the new unwind table are associated with more than one unwind table.~~

when basic blocks identified in the new unwind table are associated with more than one unwind table, creating a new region header describing a region of zero length and copying non-when action descriptor records from the original unwind descriptor records into the new unwind descriptor records.